Receiver-Initiated Channel-Hopping for Ad-Hoc Networks

Asimakis Tzamaloukas and J.J. Garcia-Luna-Aceves
Computer Engineering Department
Baskin School of Engineering
University of California, Santa Cruz, California 95064
{jamal, jj}@cse.ucsc.edu

Abstract—
The medium-access control (MAC) protocols for wireless networks proposed or implemented to date based on collision-avoidance handshakes between sender and receiver either require carrier sensing or the assignment of unique codes to nodes to ensure that intended receivers hear data packets without interference from hidden sources (i.e. IEEE 802.11). We present and analyze a receiver-initiated channel-hopping (RICH) protocol, which is the first MAC protocol based on a receiver-initiated collision-avoidance handshake that does not require carrier sensing or the assignment of unique codes to nodes to ensure collision-free reception of data at the intended receivers in the presence of hidden terminals. The correct floor acquisition for RICH is verified, and the throughput and delay characteristics are calculated analytically. The RICH protocol presented here is applicable to ad-hoc networks based on commercial, off-the-shelf, spread spectrum frequency-hopping radios operating in unlicensed frequency bands.

I. Introduction

Many medium-access control (MAC) protocols for ad-hoc networks based on collision avoidance have been proposed over the past few years. In the traditional collision-avoidance protocols, a node that needs to transmit data to a receiver first sends a request-to-send (RTS) packet to the receiver, who responds with a clear-to-send (CTS) if it receives the RTS correctly. A sender transmits a data packet only after receiving a CTS successfully. Several variations of this scheme have been developed since SRMA (split-channel reservation multiple access) was first proposed by Kleinrock and Tobagi [15], including MACA [12], MACAW [2], IEEE 802.11 [1], and FAMA [4]. More recently, receiver-initiated collision-avoidance protocols have also been proposed for single-channel networks, in which the receiver initiates the collision-avoidance handshake [6], [14]; these receiver-initiated collision-avoidance protocols also require carrier sensing to ensure correct collision avoidance.

The need for collision-avoidance MAC protocols for single-channel networks to sense the channel as an integral part of the collision-avoidance handshake limits their applicability. Some commercial radios do not provide true carrier sensing, and direct sequence spread-spectrum (DSSS) radios may capture none or one of multiple overlapping transmissions in a non-deterministic manner, depending on the proximity and transmission power of the sources. Even if frequency-hopping spread-spectrum (FHSS) radios are used, carrier sensing adds to the complexity of the radio, which must already provide coarse time synchronization at the dwell-time level. On the other hand, using one or more busy tones to indicate when a receiver is busy [9] requires, in essence, a second transceiver, which is not economically attractive.

In the past, several MAC protocols have been proposed and analyzed to take advantage of spreading codes for multiple access. Sousa and Silvester [13] presented and analyzed various spreading-code protocols that are sender-, receiver- or sender-receiver based, i.e., in which codes are assigned to senders, receivers, or combinations. Gerakoulis et. al. [7] used carrier sensing to propose a receiver-based, asynchronous transmissions protocol. Jiang and Hsiao [10] proposed a receiver-based handshake protocol for CDMA (code division multiple access) networks that improved the efficiency of the network by reducing the amount of unsuccessful transmissions and unwanted interference. Several other proposals have been made to implement correct collision-avoidance in multihop wireless networks without requiring nodes to use carrier sensing; these proposals rely on multiple codes assigned to senders or to receivers to eliminate the need for carrier sensing (e.g., [3], [5], [11]).

The key limitation of protocols based on code assignments is that senders and receivers have to find each others' codes before communicating with one another. Most of the commercial DSSS radios today use only 11 chips per bit; therefore, CDMA is not an option. Future DSSS are expected to use 15 chips per bit, allowing two different systems to operate over the same DS frequency channels as they were defined in IEEE 802.11 [1]. On the other hand, up to 26 FHSS radios can be co-located. According to the FCC regulations, up to 15 FHSS radios can be co-located with minimum interference problems. It is clear that in ad-hoc networks built with commercial radios operating in ISM bands, code assignments do not guarantee that receivers can capture one of multiple simultaneous transmissions, and that slow frequency hopping (with one or more packets sent per hop) is the viable way to achieve multiple orthogonal channels in the ISM bands. Therefore, it becomes imperative to develop MAC protocols that can take advantage of the characteristics of FHSS radios operating in ISM bands to ensure that transmissions are free of collisions due to hidden terminal interference.

Section II describes the operation of a receiver-initiated collision-avoidance (RICH) protocol that does not require code assignments or carrier sensing. RICH is based on requiring all nodes in a network to follow a common channel-hopping sequence. This requirement can be easily met in practise. A channel can be defined to be a frequency hop, a spreading code, or a combination of both. However, with commercial radios operating in ISM bands, a channel should be viewed as a frequency hop or a hopping sequence. At any given time, all nodes that are not sending or receiving data listen on the common channel hop. To send data, nodes engage in a receiver-initiated dialogue over the channel hop in which they are at the time they require to send data; those nodes that succeed in a collision-avoidance handshake remain in the same channel hop for the duration of their data transfer, and the rest of the nodes continue to follow the common channel hopping sequence. Section III proves that, in the absence of fading, RICH solves the hidden-terminal problem, i.e., it eliminates collisions of data packets, without the need for carrier sensing or code assignments. As such, RICH is the first approach reported to date that accomplish correct collision avoidance without carrier sensing or code assignment. Section IV analyzes the throughput of RICH for the case in which a single data packet is sent with every successful collision-avoidance handshake. We compare RICH with the MACA-CT protocol [11], which uses MACA collision-avoidance handshakes over a common channel and a transmitter-oriented data channel assigned to avoid collisions of data packets; we chose MACA-CT for our comparison, because it is

This work was supported by the Defense Advanced Research Projects Agency (DARPA) under Grant No. F30602-97-2-0338.

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **2000**	2. REPORT TYPE	3. DATES COVERED **00-00-2000 to 00-00-2000**
4. TITLE AND SUBTITLE **Receiver-Initiated Channel-Hopping for Ad-Hoc Networks**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California at Santa Cruz,Department of Computer Engineering,Santa Cruz,CA,95064**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR’S ACRONYM(S)
		11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**6**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

the best representative of collision-avoidance solutions that eliminate the need for carrier sensing at the expense of requiring unique channel (code) assignments. Section V calculates the system delay for RICH. Section VI summarizes our conclusions.

II. RECEIVER-INITIATED CHANNEL-HOPPING

A. Basic Concepts in Channel Hopping

RICH is based on three basic observations. First, as it has been shown [6], reversing the collision-avoidance handshake (i.e., making the receiver in charge of avoiding collisions), improves the throughput of the network. Second, hidden-terminal interference can be eliminated by the assignment of channels or codes to senders or receivers in a way that no two senders or receivers share the same code if they are two hops away from one another. Third, with commercial frequency-hopping radios operating in ISM bands, radios have to synchronize in time so that all radios hop to different frequency hops at approximately the same time.

To eliminate hidden-terminal interference, RICH exploits the fact that the nodes of a frequency-hopping network must agree on when to hop. A common frequency-hopping sequence is assumed by all the nodes (i.e., a common channel), so that nodes listen on the same channel at the same time, unless instructed otherwise. Nodes then carry out a receiver-initiated collision-avoidance handshake to determine which sender-receiver pair should remain in the present hop in order to exchange data, while all other nodes that are not engaged in data exchange continue hopping on the common hopping sequence. Because the collision-avoidance handshake ensures that the receiver of a successful handshake cannot receive packets that suffer from hidden-terminal interference, and because all nodes not able to exchange data must hop to the next frequency hop, RICH eliminates the need for carrier sensing and code assignment by simply allowing the sender and receiver of the handshake to remain on the same frequency hop in which they succeeded in their handshake.

The dwell time for a frequency hop in RICH need be only as long as it takes for a handshake to take place; as it will be clear, this time need only be long enough to transmit a pair of MAC addresses, a CRC, and framing. On the other hand, according to FCC regulations, a frequency-hop radio can remain in the same hop for up to 400msec, which at a data rate of 1 Mbps is ample time to transmit entire data packets and packet trains. Hence, RICH can be implemented by allowing a sender-receiver pair to communicate in the same frequency hop for a period of time that must be the smaller of 400msec and the time elapsed before the same frequency hop is used again in the common hopping sequence. Alternatively, a few orthogonal frequency-hopping sequences can be defined (e.g., 10, which is smaller than the number of simultaneous orthogonal frequency hops around a receiver in the 2.4 GHz band) for each frequency hop of the common hopping sequence.

B. RICH

The RICH protocol is based on simple polling by the receiver. The idea of simple polling was first introduced in MACA-BI [14] for single-channel networks and modified in RIMA [6] for correct collision avoidance over single-channel networks.

All the nodes follow a common channel-hopping sequence and each hop lasts the amount of time needed for nodes to receive a collision-avoidance control packet from a neighbor. A node attempts to poll its neighbors at a rate that is a function of the data rate with which it receives data to be sent, as well as the rate with which the node hears its neighbors send control and data packets. A node ready to poll any of its neighbors sends a ready-to-receive (RTR) control packet over the current channel hop specifying the address of the intended sender and the polling node's address. If the RTR is received successfully by the polled node, that node starts sending data to the polling node immediately and over the same channel hop, and all other nodes hop to the next channel hop. In practice, the dwell time of a channel hop needs to be only long enough to allow an RTR to be received by a polled node. When the transmission of data is completed, then sender and receiver re-synchronize to the current channel hop. If either multiple RTRs are sent during the same channel hop, or the polled node has no data to send to the polling node, the polling node does not receive any data a round-trip time after sending its RTR and must rejoin the rest of the network at the current channel hop. To permit the polling node to determine quickly that no data packet is to be expected, the polled node can transmit a short preamble packet in front of the data packet. To simplify our description, in the rest of this paper we simply assume that a node is able to detect that no data packet is arriving.

Fig. 1 illustrates the operation of RICH for the case in which sender-receiver pairs exchange data over a single frequency hop. In the figure, all the nodes start at time $t1$ from hop $h1$. At time $t2$ the system is at hop $h2$ and so on. At time $t1$ node x sends an RTR to node y and node y responds with data over the same channel. Notice that, there is a probability of $\frac{1}{N-1}$ that node y has data for x, where N is the number of nodes in the network. While x and y, stay in $h1$ until y has finished sending its data, all the other nodes hop to $h2$. At time $t2$ another node z sends an RTR to node w, but now it is the case that w does not have a data packet for z; therefore, w sends a CTS enabling z to send any data to w. At time $t4$ node z starts sending its data to w. Again, nodes z and w stay in $h2$ until z finishes sending its data, while the other nodes hop to $h3$. At time $t3$, node a sends an RTR to node b but node b is busy transmitting data to another node (uni-directional radios). Therefore, node b does not receive the RTR and at time $t4$ there is silence. In this case, node a continues to hop with the other nodes to hop $h4$. At time $t4$ nodes c and d send an RTR and therefore a collision occurs. Both nodes have to back off and try to send an RTR at a later time.

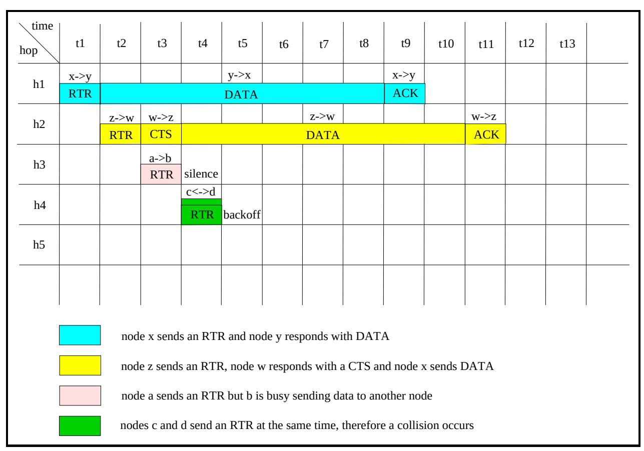

Fig. 1. RICH illustrated

After a node is properly initialized, it transitions to the PASSIVE state. In all the states, before transmitting anything to the channel, a node must listen to the channel for a period of time equal to a dwell time (time spent in one frequency hop). If node x is in PASSIVE state and obtains an outgoing packet to send to neighbor z, it transitions to the RTR state. In the RTR state, the node sends an RTR packet with the destination address of the node that is the target destination, in this case z.

If node z receives the RTR correctly and has data for x, node z transitions to the XMIT state, where it transmits a data packet to x in the same frequency hop; otherwise, if node z cannot decode the RTR correctly, it perceives noise or silence, depending on the radio being used in that hop and continues to hop with the rest of the nodes in the common hopping sequence. After sending its RTR, node x waits until the beginning of the next hop. At this time, if a preamble is not detected node x transitions to a new frequency channel according to the common hopping sequence; otherwise, x remains in the same frequency channel until (a) either a data packet arrives with the duration of it being part of its header, or (b) a Clear To Sent (CTS) packet arrives allowing x to sent a data packet at the same unique frequency channel.

When multiple RTRs are transmitted within a one-way propagation delay a collision takes place and the nodes involved have to transition to the BACKOFF state and try again at a later time chosen at random. After sending its RTR, node x waits for a response in the new frequency base. Node x determines that its RTR was not received correctly by z after a time period equal to one hop. If that is the case, node x will synchronize with the other nodes at a frequency hop that can be determined easily since node x is aware of the base frequency hop that the whole system is hopping at, from the initialization that took place at the beginning of the *hop cycle*.

To reduce the probability that the same nodes compete repeatedly for the same receiver at the time of the next RTR, the RTR specifies a back-off-period unit for contention. The nodes that must enter the BACKOFF state compute a random time that is a multiple of the back-off-period unit advertised in the RTR. The simplest case consists of computing a random number of back-off-period units using a uniformly distributed random variable from 1 to d, where d is the maximum number of neighbors for a receiver. The simplest back-off-period unit is the time it takes to send a small data packet successfully.

Although the 400msec allowed per dwell time by the FCC is a long time to transmit data in ISM bands, it may be desirable to allow nodes sending data to hop over multiple frequency hops, because staying at the same frequency hop for a long period of time does not take advantage of many inherent advantages that come with frequency hopping. For example, frequency hopping can continue to work efficiently even in the presence of narrow-band jamming, is resilient against fading and erasures, and minimizes the multi-path propagation problem. However, in order to realize these benefits, the rate with which the nodes in the network hop from one frequency to another should not be below a certain threshold.

III. Correct Collision Avoidance in RICH

Theorem 1 below shows that RICH ensures that there are no collisions between data packets and any other transmissions under the following assumptions [4]:

- *A0)* A node transmits an RTR that does not collide with any other transmissions with a non-zero probability.
- *A1)* The maximum end-to-end propagation time in the channel is $\tau < \infty$.
- *A2)* A packet sent over the channel that does not collide with other transmissions is delivered error free with a non-zero probability.
- *A3)* All nodes execute RICH correctly.
- *A4)* The transmission time of an RTR and a CTS is γ, the transmission time of a data packet is δ, and the hardware transmit-to-receive transition time is zero; furthermore, $2\tau < \gamma \leq \delta < \infty$.
- *A5)* The dwell time in each hop is equal to the time needed to transmit an RTR (or CTS) plus the maximum end-to-end propagation time.
- *A6)* There is no capture, erasure, or fading in the channel.
- *A7)* Any overlap of packet transmissions at a particular receiver, causes that receiver to not understand any of the packets.

The approach used to show that a collision-avoidance protocol works correctly, i.e., that it prevents data packets from colliding with any type of packets, consists of showing that, once a data packet is sent by a node, the intended receiver obtains the packet without interference from any other source. The intuition why this is possible is shown in Fig. 2, which illustrates that pairs of nodes can exchange data over a given hop h_i while the other nodes move on with the common hopping sequence or are exchanging data over a different hop.

Theorem 1: RICH provides correct collision avoidance in the presence of hidden terminals when the time spent exchanging data is shorter than the time elapsed before the same frequency hop is reused in the common hopping sequence.

Proof: Consider a polling node A and a polled node X and assume that A sends an RTR at time t_0. After sending its RTR, node A remains in frequency hop H for a period of time that is long enough to detect a CTS or the presence or absence of a data packet. We denote by h the dwell time in a particular hop. If X does not receive the RTR correctly due to interference from any neighbor hidden from A, it does not send any data. Else, X receives A's RTR at time $t_1 = t_0 + h$ and remains in the same frequency hop H where the RTR was received. At time $t_1' > t_0 + h$, if node X has a local data packet for A, then it starts sending its data to A; otherwise, X sends a CTS to A enabling A to send its data packet. Both nodes A and X remain in frequency hop H, that never collides with the common hopping sequence since we made the assumption that the time spent exchanging data is shorter than the time elapsed before the same frequency hop is reused in the common hopping sequence (Fig. 2). □

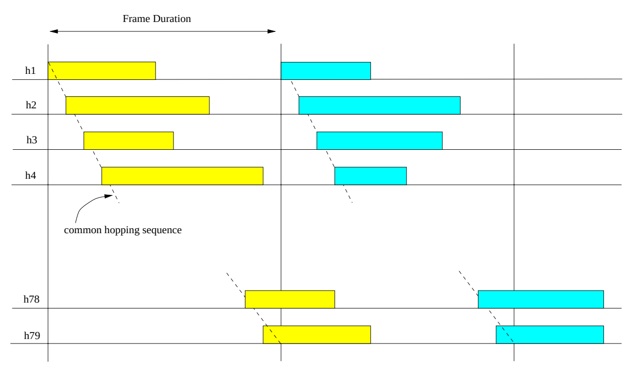

Fig. 2. RICH provides correct floor acquisition since there are no conflicts between the common frequency hopping sequence and ongoing DATA packet transmissions

IV. Approximate Throughput Analysis

The objective of our analysis is to calculate the throughput achieved with RICH, and to compare that against sender-initiated CDMA protocols such as MACA-CT [11]. The choice of protocols was made because we wanted to show how RICH performs against the best performing sender-initiated protocols reported to date for ad hoc networks in which receivers can detect at most one transmissions at a time. Our analysis shows that RICH has a much better performance than MACA-CT.

A. Assumptions

We analyze the throughput of receiver initiated protocols using the model first introduced by Sousa and Silvester [13] for CDMA protocols. We calculate the throughput and average delay for RICH with a discrete-time Markov chain. The following assumptions are made:

1. There are N nodes in the network.
2. A single unslotted channel is used for all packets, and the channel introduces no errors.
3. At any given time slot, at most one RTR can be successfully transmitted.
4. All nodes can detect collisions perfectly; there is no capture or fading.
5. The data packet length distribution is geometrically distributed with parameter q; therefore, the probability of a data packet with length l is, $P[L = l] = (1-q)q^{l-1}$ and the average packet length, measured in minipackets per slot is, $\overline{L} = \frac{1}{1-q}$.

A polled node has a packet addressed to the polling node with probability $\frac{1}{N-1}$ (i.e. uniform distribution). Furthermore, we assume that each node sends its RTR according to a Poisson distribution with a mean rate of $\frac{\lambda}{N-1}$, and that (when applicable) the polling node chooses the recipient of the RTR with equal probability.

B. RICH

To make a fair comparison with MACA-CT, we use the same average packet length, L, in all protocols. However, since in MACA-CT a slot is equal to the size of an RTS plus a CTS plus the corresponding propagation time needed, the duration of a slot size, h, for RICH protocols is equal to half the size of the slots used in MACA-CT. Consequently, the average packet length for MACA-CT will be equal to $\frac{1}{2(1-q)}$.

At any given slot, a node can be: (a) idle, (b) transmitting an RTR or a CTS control signal, and (c) sending a series of consecutive (in time) slots with segments of the data packet. The possible scenarios that can occur in RICH are:

- node x sends an RTR to node y and y sends its data packet to x with probability $\frac{1}{N-1}$
- node x sends an RTR to node y but y does not have any data for x, therefore y sends a CTS to x and x sends its data to y
- node x sends an RTR at the same time that node y sends an RTR, therefore a collision occurs
- node x sends an RTR but node y is already tuned in a different hopping pattern, therefore node x does not hear anything in the next hop

At any given time the system state can be described by the number of communicating pairs of nodes (Fig. 3). Notice that, since all the nodes that transmit an RTR that is not received at time slot $t-1$ are available at slot t, the system state at any given time slot t is independent from the number of nodes that send an unanswered RTR. Accordingly, we need to calculate the transition probabilities of this Markov chain under the assumptions presented above. A transition in the Markov chain from one state to another occurs when: (a) at least one member from the set of nodes exchanging data packets, finish transmitting data, and (b) the nodes that participate in the handshake either succeed or fail sending an RTR. To calculate the transition probability from the current state we need to know the number of nodes that will finish sending data and the number of nodes that succeed or fail sending an RTR.

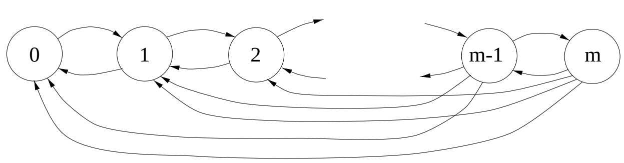

Fig. 3. Markov Chain defining the average number of communicating pairs

Since we have made the assumption of geometrically distributed data packet lengths with parameter q, the probability that i pairs of nodes will become idle in any given time slot is equal to [13]

$$Pr\begin{pmatrix} i \text{ pairs} \\ \text{become idle} \end{pmatrix} = \binom{m}{i}(1-q)^i q^{(m-i)} \qquad (1)$$

Let $P_{k,l}$ be the transition probability in the Markov chain from state k (where k pairs of nodes exchange data) in slot $t-1$, to state l (where l pairs of nodes exchange data) in slot t. We condition on the number i of communicating pairs of nodes that finish sending or receiving data packets at the beginning of slot t. The system is at state l at time slot $t-1$ and therefore the number of nodes that are available to receive or transmit is equal to $N-2(l-i)$. If the transition to state l is made, then let x' be the number of nodes which transmit an RTR at the beginning of time slot t. Furthermore, $l' = l-(k-i)$ pairs of nodes will become busy exchanging data packets and $n' = x' - l'$ nodes will transmit an RTR packet that will not be received. Due to the assumption that only one RTR can be successful at a given time slot, a transition from state k to state l is possible only if $m' = 1$ or $m' = 0$.

We denote with Φ the event that a transition from k to l occurs, with ΦI being the event that exactly one transmission occurs and it is addressed to an idle node, and with ΦB being the event that exactly one transmission occurs and it is addressed to a busy terminal. Then, the transition probabilities can be calculated as follows [13]

$$p_{k,l} = q^{l-1}(1-q)^{k-1}\left\{\binom{k}{l-1}(1-q)p(1-p)^{M+1}\frac{M^2+3M+2}{N-1} - \binom{k}{l}qp(1-p)^{M-1}\frac{M^2-M}{N-1} + \binom{k}{l}q\right\} \qquad (2)$$

where $M = N-2l$. To calculate the average throughput we need to know the steady-state probabilities that correspond to each one of the states of the Markov chain (Fig. 3). From the transition probability equation, we can solve a linear system of equations with as many unknowns as the number of states in the Markov chain to calculate the steady-state probabilities. If PS_l is the steady state probability for state l, then the average throughput S is equal to the number of data packets transmitted at the same frequency hop; that is

$$S = \sum_{l=0}^{\frac{N}{2}} l \cdot PS_l \qquad (3)$$

Figure 4 shows the throughput achieved by RICH and MACA-CT versus the probability of transmission p for various numbers of nodes in the network. Because the slot duration in RICH is half the one in MACA-CT, the probability of transmission at a given slot is $\frac{p}{2}$. The maximum throughput of RICH is always higher than MACA-CT because the duration for the exchange of the control signals is half the size of the one used in MACA-CT and consequently the vulnerability period in RICH is half the time spent in MACA-CT. Since no data will be ever send with RICH to a busy terminal, nodes in RICH are immediately available to try again, something that is not the case in C-T [13]. Therefore, at any given time slot, the number of nodes available to transmit an RTR is maximized while the contention period is minimized!

Figure 5 shows the throughput against the probability of transmission p for a fixed number of nodes ($N = 12$) with the average packet length L being the parameter. As it is obvious, RICH has a higher throughput than MACA-CT regardless of the size of the data packet. The general conclusion that can be drawn in this case is that, higher throughput can be achieved with a longer average packet length. However, notice that we have made the assumption of a perfect channel. In a realistic environment, by increasing the length of the transmitted packet we also increase the probability that errors will occur. Furthermore, when the number of co-located nodes is high, the interference from adjacent frequency channels is more likely to introduce errors in the transmission of data packets. It has been shown [8] that there is no improvement in the throughput achieved by increasing the length of the data packet after a certain threshold in a non-perfect channel for other spread spectrum protocols. The same should be expected for RICH.

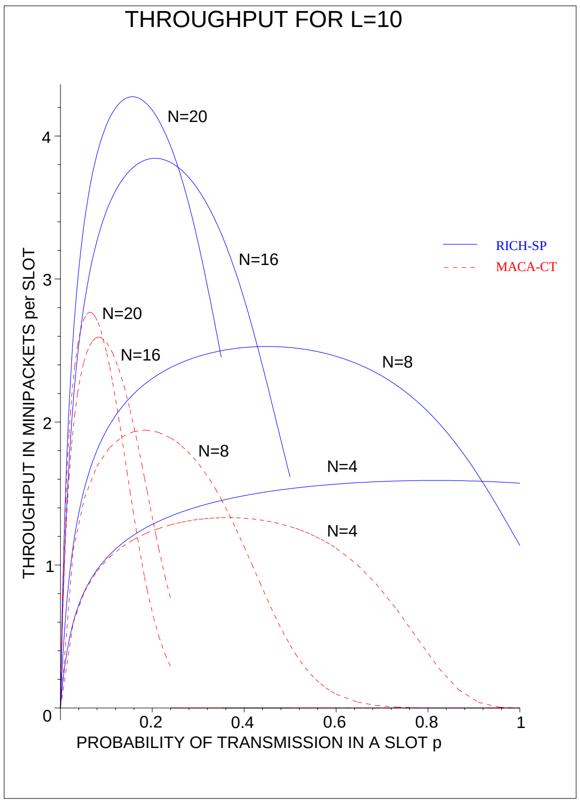

Fig. 4. Throughput versus transmission probability for MACA-CT and RICH for a fixed average packet length $\overline{L} = 10$

V. Delay Analysis

To calculate the average delay for RICH we need to first define a retransmission policy. We assume that the arrival process is Bernoulli with probability p for every node. Because we have a queue of maximum size equal to one packet, if a packet is waiting in the queue then there are no further new packet arrivals, and the waiting packet is retransmitted in the next slot with probability p. If a node has a packet waiting to be sent, but a packet from some other user is received, then the waiting packet is discarded and when the handshake is completed the given node becomes idle and generates a new packet with probability p. All the assumptions that were presented in section IV are valid in the following derivation as well.

We use Little's theorem to calculate the average delay. We define the system delay D as the time that it takes for a new arriving packet that is waiting in the queue to be transmitted and successfully received by the intended receiver. If $\overline{m}$ is the average number of pairs of nodes that simultaneously exchange data packets, and $\overline{B}$ is the average number of blocked users (due to collision of RTSs or RTSs that are not received), then at any given time the average number of packets in the system will be equal to $\overline{m} + \overline{B}$. We can calculate $\overline{m}$ and $\overline{B}$ as follows

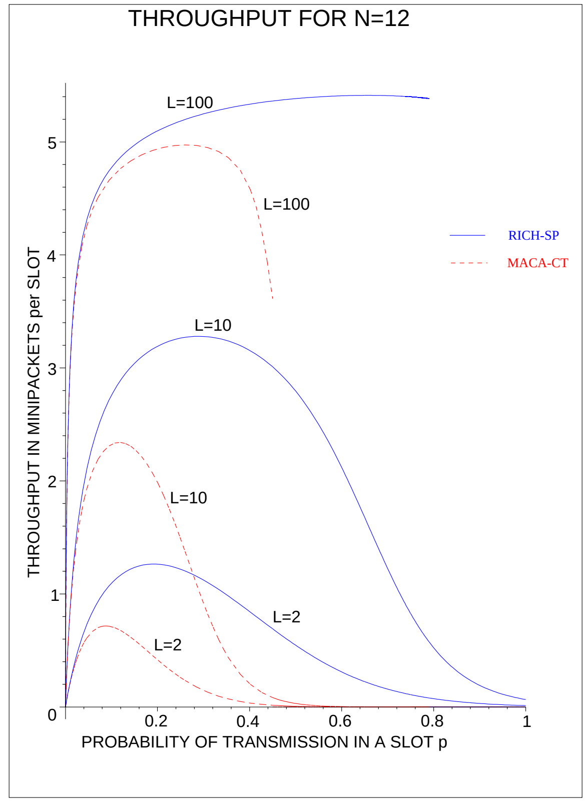

Fig. 5. Throughput versus transmission probability for MACA-CT and RICH for a fixed number of nodes $N = 12$

$$\overline{m} = \sum_{m=0}^{\lfloor \frac{N}{2} \rfloor} m P_m \tag{4}$$

and

$$\overline{B} = \sum_{m=0}^{\lfloor \frac{N}{2} \rfloor} p(N-2m)\left(1 - \frac{N-m-1}{N-1}\right) P_m \tag{5}$$

The average delay normalized to a packet length is derived by applying Little's theorem as follows

$$\overline{D} = \frac{\overline{m} + \overline{B}}{S} \tag{6}$$

Since the mean transmission time for a packet is equal to $\frac{1}{1-q}$ the actual system delay should include the transmission time for the data packet. That is

$$D = \frac{\overline{D}}{(1-q)} \tag{7}$$

In Figure 6 we can see the numerical results obtained for the normalized delay performance of MACA-CT and RICH. It is clear that RICH offers the smallest delay at any load. Furthermore, the system delay with RICH remains almost the same up to $p > 0.6$ whereas with MACA-CT the delay increases exponentially when $p > 0.4$. This is to be expected, because collisions between control packets increase as the offered load increases, and minimizing the length of the collision-avoidance handshakes that are susceptible to collisions becomes crucial. Indeed, with RICH, only RTRs can collide and therefore their vulnerability periods are half the vulnerability period in MACA-CT. It is obvious from the same figure the normalized delay can be reduced noticeably by increasing the packet length.

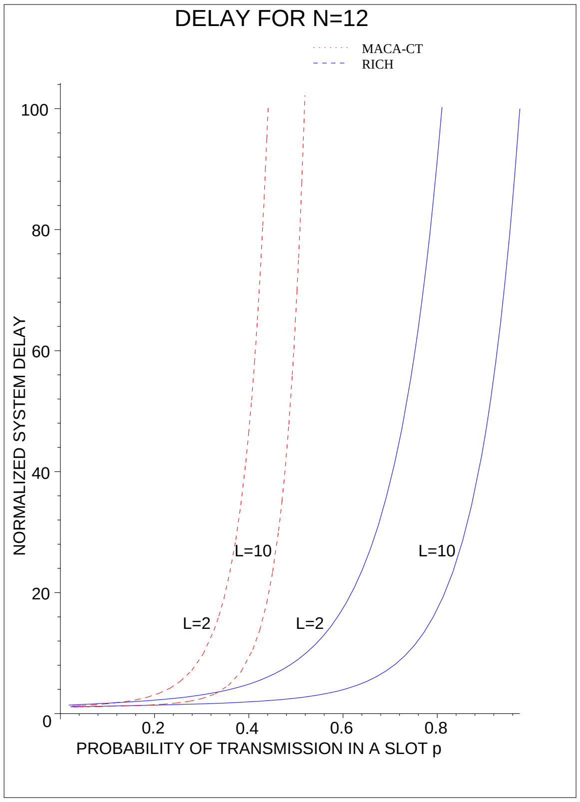

Fig. 6. Normalized system delay versus transmission probability for MACA-CT and RICH for a fixed number of nodes $N = 12$

In Figure 7 the actual system delay that includes the packet transmission time is shown. In this figure, contrary to what happened with the normalized system delay, we notice that by increasing the packet length we do not achieve smaller delays. However, this is to be expected since the transmission time is the dominating delay in this case.

VI. Conclusions

We have presented a family of receiver-initiated collision-avoidance protocols that correctly eliminate hidden-terminal interference without the need for carrier sensing or the assignment of unique codes to network nodes, both of which are difficult to accomplish in ad-hoc networks based on commercial radios operating in ISM bands. We proved that RICH does eliminate hidden-terminal interference and compared their throughput against MACA-CT, which is a recent example of collision-avoidance protocols that do not require carrier sensing but need code assignment to operate correctly. For this comparison, we used the same analysis method introduced by Sousa and Silvester for code-hopping protocols [13] and showed that RICH achieves higher throughput than MACA-CT, without the need for any code assignments. Various simulation scenarios were developed to verify the analysis.

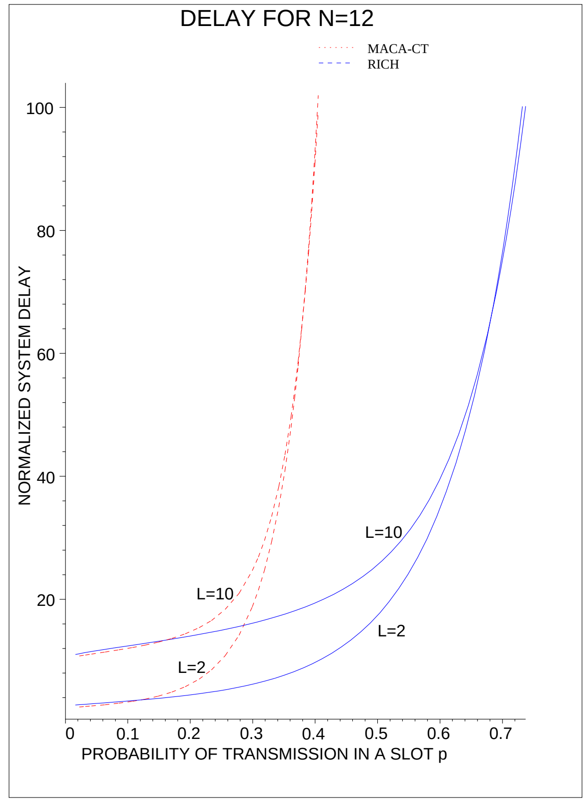

Fig. 7. Actual system delay versus transmission probability for MACA-CT and RICH for a fixed number of nodes $N = 12$

References

[1] *IEEE P802.11 Draft Standard for Wireless LAN: Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE, July 1996.

[2] Bharghavan, V. and Demers, A. and Shenker, S. and Zhang, L. MACAW: A Media Access Protocol for Wireless LAN's. In *Proceedings ACM SIGCOMM*, pages 212–25, London, UK, August 1994.

[3] C. Fullmer. *Collision Avoidance Techniques for Packet radio Networks*. Ph.D. Thesis, UC Santa Cruz, 1998.

[4] Fullmer, C. L. and Garcia-Luna-Aceves, J J. Solutions to Hidden Terminal Problems in Wireless Networks. In *Proceedings ACM SIGCOMM*, Cannes, France, September 1997.

[5] R. Garces. *CARMA: Collision Avoidance and Resolution Multiple Access*. Ph.D. Thesis, UC Santa Cruz, 1999.

[6] J. J. Garcia-Luna-Aceves and A. Tzamaloukas. Reversing the Collision-Avoidance Handshake in Wireless Networks. In *Proceedings ACM/IEEE MobiCom '99*, Seattle, Washington, August 1999.

[7] Saadawi T. N. Gerakoulis, P. D. and D. L. Schilling. A Channel Access Protocol for Embedding CSMA on Spread Spectrum Packet Radio Networks. *Proceedings of IEEE ICC*, 1988.

[8] E. Geraniotis and M. Pursley. Error Probability for Direct Sequence Spread Spectrum Multiple Access Communication - Part II:Approximations. *IEEE Transactions on Communications*, COM-30, May 1982.

[9] Z. Haas and J. Deng. Dual Busy Tone Multiple Access (DBTMA): a New Medium Access Control for Packet Radio Networks. In *IEEE 1998 International Conference on Universal Personal Communications*, Florence, Italy, October 1998.

[10] S. Jiang and T. Hsiao. Performance Evaluation of a Receiver-Based Handshake Protocol for CDMA Networks. *IEEE Transactions on Communications*, 43, 1995.

[11] M. Joa-Ng and I. Lu. Spread Spectrum Medium Access Protocol with Collision Avoidance in Mobile Ad-Hoc Wireless networks. In *Proc. IEEE INFOCOM 99*, April 1999.

[12] P. Karn. MACA - a New Channel Access Method for Packet Radio. In *Proceedings ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, New York, April 1990.

[13] E. S. Sousa and J. A. Silvester. Spreading Code Protocols for Distributed Spread Spectrum Packet Radio Networks. *IEEE Transactions on Communications*, 36, March 1988.

[14] F. Talucci, M. Gerla, and L. Fratta. MACA-BI (MACA by invitation) - A Receiver Oriented Access Protocol for Wireless Multihop Networks. In *Proceedings IEEE PIMRC*, 1997.

[15] F. A. Tobagi and L. Kleinrock. Packet Switching in Radio Channels: Part III - Polling and (dynamic) Split Channel Reservation Multiple Access. *IEEE Transactions on Computers*, 24(7):832–45, August 1976.